AMENDMENTS TO THE CLAIMS

1. (currently amended) A method for preparing a logic structure for random pattern testing, the method comprising:

configuring a select mechanism within a data scan chain, said select mechanism configured between a first register in said data scan chain and a second register; and

routing a parallel data path within said scan chain, said parallel data path beginning from an input side of said first register, running through said select mechanism, and ending at an input side of said second register;

said select mechanism being capable of switching a source path of input data to said second register ~~from~~ between a normal data path ~~to~~ and said parallel data path;

wherein, when said parallel data path is selected as said source path of input data to said second register, data loaded from said parallel data path into said second register ~~matches~~ is selectively identical to data loaded into said first register from said input side of said first register.

2. (original) The method of claim 1, wherein said first and second registers contain an equal number of data storage elements therein.

3. (currently amended) The method of claim 1, further comprising:

selecting said parallel data path as said source path of input data to said second register; and

inputting the contents of said first and second registers into the logic structure for testing.

4. (currently amended) The method of claim 1, further comprising:

configuring a bitflip logic mechanism within said parallel data path, said

bitflip logic mechanism capable of inverting one or more data bits passing through said parallel data path;

wherein, when said parallel data path is selected as said source path of input data to said second register and said bitflip logic mechanism is activated, said data loaded into said second register is ~~may be~~ statistically biased to be mismatched from said data loaded into said first register by one bit or more.

5. (original) The method of claim 4, further comprising:

configuring weight logic to control a frequency of occurrences in which said bitflip logic mechanism is caused to invert said one or more data bits passing through said parallel data path.

6. (original) The method of claim 5, wherein:

said weight logic further comprises a multiple-input AND gate, each of said multiple inputs being coupled to independent, random bit generating devices.

7. (original) The method of claim 6, wherein:

said bitflip logic mechanism further comprises an exclusive OR (XOR) gate, said XOR gate having an output of said multiple-input AND gate as a first input thereto, and a corresponding data bit in said parallel data path as a second input thereto.

8. (cancelled)

9. (currently amended) ~~The method of claim 8, further comprising:~~ A method for configuring a built in, self-test (BIST) circuit used in random pattern testing of integrated circuits, the BIST circuit including a first register therein, and plurality of subsequent registers for storing data to be tested with data contained in the first register, the method comprising:

configuring a plurality of select mechanisms, each of said plurality of

select mechanisms corresponding to one of the subsequent registers;

routing a plurality of parallel data paths within a scan chain containing the first register therein, said plurality of parallel data paths each beginning from an input side of said first register, running through a corresponding one of said plurality of select mechanisms, and ending at an input side of a corresponding one of the subsequent registers;

said plurality of select mechanisms each being capable of switching a source path of input data to each of the subsequent registers ~~from~~ between a normal data path ~~to~~ and a corresponding one of said parallel data paths;

wherein, for each of the subsequent registers having one of said parallel data paths selected as said source path of input data thereto, the data loaded therein from said corresponding one of said parallel data paths ~~matches~~ is selectively identical to data loaded into the first register from said input side of said first register.

10. (currently amended) The method of claim ~~8~~9, wherein the first register and the subsequent registers contain an equal number of data storage elements therein.

11. (currently amended) The method of claim 9, further comprising:

configuring a bitflip logic mechanism within each of said plurality of parallel data paths, said bitflip logic mechanisms capable of inverting one or more data bits passing through said plurality of parallel data paths;

wherein, whenever one of said plurality of parallel data paths is selected as said source path of input data to a corresponding one of the subsequent registers, and a corresponding bitflip logic mechanism thereto is activated, said data loaded into said corresponding one of the subsequent registers ~~may be~~is statistically biased to be mismatched from said data loaded into the first register by one bit or more.

12. (original) The method of claim 11, further comprising:

configuring weight logic to control a frequency of occurrences in which

each of said bitflip logic mechanisms is caused to invert said one or more data bits passing through said plurality of parallel data paths.

13. (original) The method of claim 12, wherein:

said weight logic further comprises a multiple-input AND gate, each of said multiple inputs being coupled to independent, random bit generating devices.

14. (original) The method of claim 13, wherein:

said bitflip logic mechanism further comprises an exclusive OR (XOR) gate, said XOR gate having an output of said multiple-input AND gate as a first input thereto, and a corresponding data bit in said plurality of parallel data paths as a second input thereto.

15. (currently amended) An apparatus for preparing a logic structure for random pattern testing, comprising:

a select mechanism configured within a data scan chain, said select mechanism coupled between a first register in said data scan chain and a second register; and

a parallel data path routed within said scan chain, said parallel data path beginning from an input side of said first register, running through said select mechanism, and ending at an input side of said second register;

said select mechanism being capable of switching a source path of input data to said second register ~~from~~ between a normal data path ~~to~~ and said parallel data path;

wherein, when said parallel data path is selected by said select mechanism as said source path of input data to said second register, data loaded into said second register from said parallel data path ~~matches~~ is selectively identical to data loaded into said first register from said input side of said first register.

16. (original) The apparatus of claim 15, wherein said first and second registers contain an equal number of data storage elements therein.

17. (currently amended) The apparatus of claim 15, further comprising:

a bitflip logic mechanism configured within said parallel data path, said bitflip logic mechanism capable of inverting one or more data bits passing through said parallel data path;

wherein, when said parallel data path is selected as said source path of input data to said second register and said bitflip logic mechanism is activated, said data loaded into said second register ~~may be~~is statistically biased to be mismatched from said data loaded into said first register by one bit or more.

18. (original) The apparatus of claim 17, further comprising:

weight logic coupled to said bitflip mechanism, said weight logic used to control a frequency of occurrences in which said bitflip logic mechanism is caused to invert said one or more data bits passing through said parallel data path.

19. (original) The apparatus of claim 18, wherein:

said weight logic further comprises a multiple-input AND gate, each of said multiple inputs being coupled to independent, random bit generating devices.

20. (original) The apparatus of claim 19, wherein:

said bitflip logic mechanism further comprises an exclusive OR (XOR) gate, said XOR gate having an output of said multiple-input AND gate as a first input thereto, and a corresponding data bit in said parallel data path as a second input thereto.